

# EE4717/IM4717 Web Application Design Table & Form

Lecturer :

Associate Professor CHONG Yong Kim

E-mail: [eykchong@ntu.edu.sg](mailto:eykchong@ntu.edu.sg)

Tel: 67904535

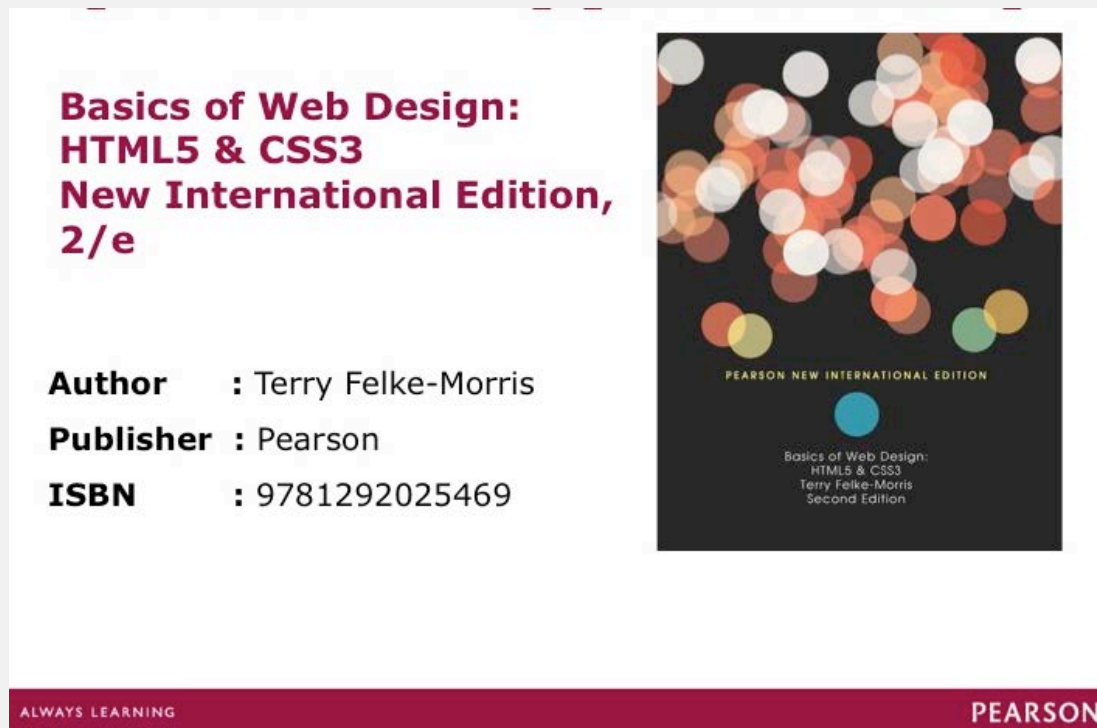


A PDF file is available for printing purpose.

No re-distribution and upload of the teaching slides, supplementary materials and recorded multimedia presentations to any publicly accessible media platform and websites.

# Copyright Notice

- The teaching slides on HTML5 and CSS3 are mostly adopted from the slides provided by the author and publisher of the following textbook:



- You should use this material strictly for your own study only. No distribution of this teaching material is allowed.

# HTML Tables

- Composed of rows and columns – similar to a spreadsheet.
  - **<table>** Element  
Contains the table
  - **<tr>** Element  
Contains a table row
  - **<td>** Element  
Contains a table data cell
  - **<th>** Element  
Contains a table header cell
  - **<caption>** Element  
Configures a description of the table

# Exercise TAB01

- Start with an HTML template, then type the following codes:

```
<table border="1">  
  <caption>Birthday List</caption>  
  <tr>    <td>Name</td>  
    <td>Birthday</td>  
  </tr>  
  <tr>    <td>James</td>  
    <td>11/08</td>  
  </tr>  
  <tr>    <td>Karen</td>  
    <td>4/17</td>  
  </tr>  
  <tr>    <td>Sparky</td>  
    <td>11/28</td>  
  </tr>  
</table>
```

Birthday List

Name	Birthday
James	11/08
Karen	4/17
Sparky	11/28

- Save it as “**TAB01.html**” and study the output in different browsers.

# Exercise TAB02

- Based on the previous exercise, change the following line to make the first row bold as shown using `<th>`:

Using the `<th>` Element

```
<tr>  <th>Name</th>
      <th>Birthday</th>
</tr>
```

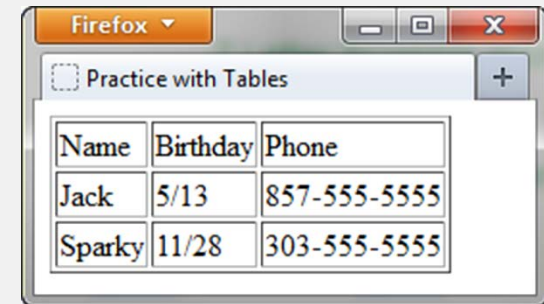
Birthday List

Name	Birthday
James	11/08
Karen	4/17
Sparky	11/28

- Save it as “**TAB02.html**” and study the output in different browsers.

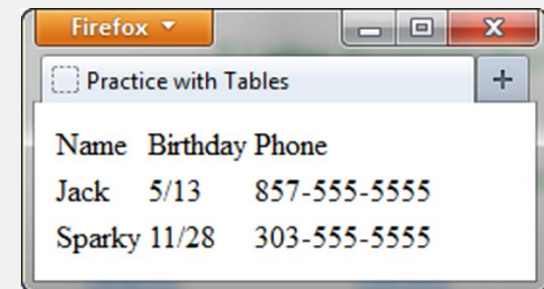
# HTML border Attribute

- Indicates the table is specifically not used for page layout
  - Optional
  - **border= "1"**  
Visible browser default border
  - **border= "0"**  
No visible browser default border.



A screenshot of a Firefox browser window titled "Practice with Tables". The window displays a table with three columns: "Name", "Birthday", and "Phone". The table has a visible border. The data rows are:

Name	Birthday	Phone
Jack	5/13	857-555-5555
Sparky	11/28	303-555-5555



A screenshot of a Firefox browser window titled "Practice with Tables". The window displays the same data as the previous table, but without visible borders. The data is as follows:

Name	Birthday	Phone
Jack	5/13	857-555-5555
Sparky	11/28	303-555-5555

# HTML colspan Attribute

- Specifies the **number of column** that a cell will occupy, e.g. 2 columns in the following example

```
<table border="1">
<tr>
  <td colspan="2"> Birthday List</td>
</tr>
<tr>
  <td>James</td>
  <td>11/08</td>
</tr>
<tr>
  <td>Karen</td>
  <td>4/17</td>
</tr>
</table>
```

Birthday List	
James	11/08
Karen	4/17

# HTML rowspan Attribute

- Specifies the **number of rows** that a cell will occupy, e.g. 2 rows in the following example:

```
<table border="1">  
  <tr>  
    <td rowspan="2">This spans two rows</td>  
    <td>Row 1 Column 2</td>  
  </tr>  
  <tr>  
    <td>Row 2 Column 2</td>  
  </tr>  
</table>
```

This spans two rows	Row 1 Column 2
	Row 2 Column 2



# Table Row Groups

```

<table>
<thead>
<tr>
  <th id="day">Day</th>
  <th id="hours">Hours</th>
</tr>
</thead>
<tbody> <tr>
  <td headers="day">Monday</td>
  <td headers="hours">4</td>
</tr> <tr>
  <td headers="day">Tuesday</td>
  <td headers="hours">3</td>
</tr> </tbody>
<tfoot> <tr>
  <td headers="day">Total</td>
  <td headers="hours">7</td>
</tr> </tfoot>
</table>

```

- **<thead>**  
table head rows
- **<tbody>**  
table body rows
- **<tfoot>**  
table footer rows

Work Schedule

Day	Hours
Monday	4
Tuesday	3
Total	7

# Exercise TAB03

---

- Try out the attributes on previous 4 slides by repeating the tables in the html file.
- Save the file as “**TAB03.html**”.

# Using CSS to Style a Table

HTML Attribute	CSS Property
align	Align a table: <code>table { width: 75%; margin: auto; }</code> Align within a table cell: <code>text-align</code>
bgcolor	background-color
cellpadding	padding
cellspacing	border-spacing or border-collapse
height	height
valign	vertical-align
width	width
border	border, border-style, or border-spacing
bg image	background-image
caption position	caption-side

# Exercise TAB04

- Type the codes without the `<style> .... </style>`. Study the output in the browser.
- Then, add the codes for `<style> ....</style>`. Study the output by refreshing the browser. Also, observe the effect in different browsers.
- Save it as “**TAB04.html**”

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<title>Table Practice</title>
```

```
<meta charset="utf-8">
```

```
<style>
```

```
table { margin: auto; border: 5px solid #000066; width: 600px; border-spacing: 0;}
```

```
td, th { padding: 5px; font-family: Arial, sans-serif; border-style: none;}
```

```
caption { font-family: Verdana, sans-serif; font-weight: bold;
```

```
font-size: 1.2em; padding-bottom: 5px; }
```

```
tr:nth-of-type(even) { background-color:#eaeaea; }
```

```
tr:first-of-type { background-color: #000066;
```

```
color: #eaeaea; }
```

```
</style>
```

```
</head>
```

**Lighthouse Island Bistro Specialty Coffee Menu**

Specialty Coffee	Description	Price
Lite Latte	Indulge in a shot of organic, locally roasted espresso with steamed, skim milk.	\$3.50
Mocha Latte	Chocolate lovers will enjoy a shot of organic, locally roasted espresso, steamed milk, and your choice of melted dark, milk, or white chocolate.	\$4.00
MCP Latte	A luscious mocha latte with caramel and pecan syrup.	\$4.50

# Exercise TAB04(cont.)

```

</body>
<table border="1">
<caption>Lighthouse Island Bistro Specialty Coffee Menu</caption>
<tr>
<th>Specialty Coffee</th>
<th>Description</th>
<th>Price</th>
</tr>
<tr>
<td>Lite Latte</td>
<td>Indulge in a shot of organic, locally roasted espresso with steamed,
skim milk. </td>
<td>$3.50</td>
</tr>
<tr>
<td>Mocha Latte</td>
<td>Chocolate lovers will enjoy a shot of organic, locally roasted espresso,
steamed milk, and your choice of melted dark, milk, or white chocolate.</td>
<td>$4.00</td>
</tr>
<tr>
<td>MCP Latte</td>
<td>A luscious mocha latte with caramel and pecan syrup.</td>
<td>$4.50</td>
</tr>
</table>
</body>
</html>

```

**Lighthouse Island Bistro Specialty Coffee Menu**

Specialty Coffee	Description	Price
Lite Latte	Indulge in a shot of organic, locally roasted espresso with steamed, skim milk.	\$3.50
Mocha Latte	Chocolate lovers will enjoy a shot of organic, locally roasted espresso, steamed milk, and your choice of melted dark, milk, or white chocolate.	\$4.00
MCP Latte	A luscious mocha latte with caramel and pecan syrup.	\$4.50

# Exercise TAB05

- Type the codes. Study the output in different browsers.
- Save it as “**TAB05.html**”

## Time Sheet

Day	Hours
Monday	4
Tuesday	3
Wednesday	5
Thursday	3
Friday	3
<b>Total</b>	<b>18</b>

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Time Sheet</title>
<meta charset="utf-8">
<style>
table { width: 200px;
        margin: auto;
    }
table, th, td { border-style: none; }
caption { font-size: 2em;
          font-weight: bold;
    }
thead {background-color: #eaeaea;
    }
tbody {font-family: Arial, sans-serif;
        font-size: .90em;
    }
tbody td { border-bottom: 1px #000033 dashed;
           padding-left: 25px;
    }
tfoot {background-color: #eaeaea;
        font-weight: bold;
        text-align: center;
    }
</style>
</head>

```

# Exercise TAB05 (cont.)

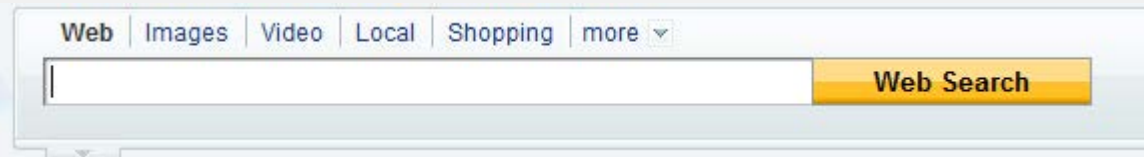
```
<body>
  <table border="1">
    <caption>Time Sheet</caption>
    <thead>
      <tr>
        <th id="day">Day</th>
        <th id="hours">Hours</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td headers="day">Monday</td>
        <td headers="hours">4</td>
      </tr>
      <tr>
        <td headers="day">Tuesday</td>
        <td headers="hours">3</td>
      </tr>
      <tr>
        <td headers="day">Wednesday</td>
        <td headers="hours">5</td>
      </tr>
      <tr>
```

```

        <td headers="day">Thursday</td>
        <td headers="hours">3</td>
      </tr>
      <tr>
        <td headers="day">Friday</td>
        <td headers="hours">3</td>
      </tr>
    </tbody>
    <tfoot>
      <tr>
        <td headers="day">Total</td>
        <td headers="hours">18</td>
      </tr>
    </tfoot>
  </table>
</body>
</html>
```

# Overview of Forms

- Types of forms:
  - Search form, Order form, Newsletter sign-up form, Survey form, Add to Cart form, and so on...



- Two Components of Using Forms

## 1. The web page form

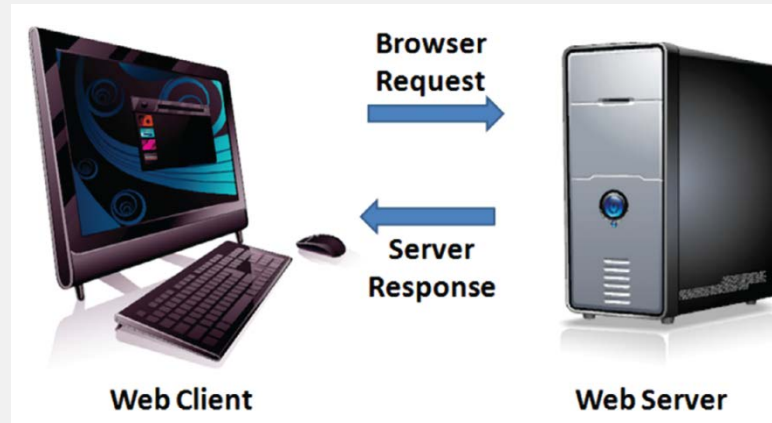
An HTML element that contains and organizes form controls such as **text boxes**, **check boxes**, and **buttons** that can accept information from website visitors.

## 2. The server-side processing

Server-side processing works with the form data and **sends e-mail**, **writes to a text file**, **updates a database**, or performs some **other type of processing** on the server.



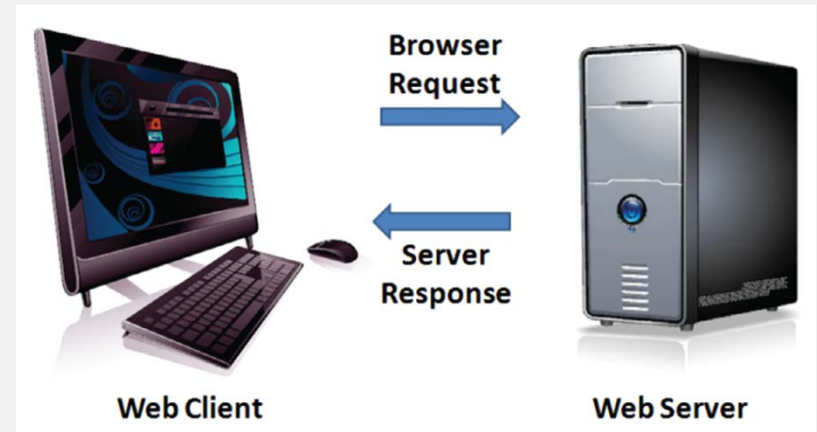
# Server-Side Processing



- Your web browser requests web pages and their related files from a web server.
- The web server locates the files and sends them to your web browser.
- The web browser then renders the returned files and displays the requested web pages for you to use.

# Server-Side Scripting

- One of many technologies in which a server-side script is **embedded** within a web page document saved with a file extension such as:
  - **.php** (PHP)
  - **.cfm** (Adobe ColdFusion)
  - **.jsp** (Sun JavaServer Pages)
  - **.aspx** (ASP.Net).
- Uses direct execution — the **script is run** either by the **web server** itself or by an extension module to the web server.



# HTML Using Forms

- **<form>** tag
  - Contains the form elements on a web page
  - Container tag
- **<input>** tag
  - Configures a variety of form elements including text boxes, radio buttons, check boxes, and buttons
  - Stand alone tag
- **<textarea>** tag
  - Configures a scrolling text box
  - Container tag
- **<select>** tag
  - Configures a select box (drop down list)
  - Container tag
- **<option>** tag
  - Configures an option in the select box
  - Container tag

# Sample Form

```
<h1>Join Our Newsletter</h1>
```

```
<form method="get">
```

```
E-mail: <input type="text" name="email" id="email" ><br><br>
```

```
<input type="submit" value="Sign Me Up!"> <input type="reset">
```

```
</form>
```



# The Form Element <form>

- The form element attributes:
  - **action**
    - Specifies the **server-side program or script** that will process your form data
  - **method**
    - **get** – default value, form data passed in URL
    - **post** – more secure, form data passed in HTTP Entity Body
  - **name**
    - Identifies the form
  - **id**
    - Identifies the form

# The Input Element Text Box <input>

- Accepts text information
- Common Attributes:
  - **type="text"**
  - name
  - id
  - size
  - maxlength
  - value
  - required (HTML5)
  - placeholder (HTML5)



A sample text box with a light gray background and a subtle drop shadow. At the top, the text "Sample Text Box" is displayed in a bold, black, serif font. Below this, the label "Email:" is followed by a rectangular text input field with a thin blue border and a light gray background.

# The Input Element Password Box <input>

- Accepts text information that needs to be **hidden** as it is entered
- Common Attributes:
  - **type="password"**
  - name
  - id
  - size
  - maxlength
  - value
  - required (HTML5)
  - placeholder (HTML5)



A sample password box is shown within a light gray rectangular frame. The frame has a title "Sample Password Box" at the top. Below the title, the text "Password:" is followed by a text input field. The input field contains ten asterisks (\*) and a vertical cursor line at the end, indicating that the text is hidden.

➤ You should use **method="post"**.

# The Input Element Check box <input>

- Allows the user to select **one or more** of a group of predetermined items

- Common Attributes:

- **type="checkbox"**
- name
- id
- **checked**
- Value

## Sample Check Box

Choose the browsers you use:

- ☐ Internet Explorer  
☐ Firefox  
☐ Opera

```
<h2>Slide 24a</h2>
<h3>checkbox</h3>
<form action="show_get" method="get">
  <input type="checkbox" name="browser" value="ie"> Internet Explore<br>
  <input type="checkbox" name="browser" value="moz" checked> Firefox<br>
  <input type="submit">
</form>

<h2>Slide 24b</h2>
<h3>checkbox multivalue</h3>
<form action="show_get" method="get">
  <input type="checkbox" name="browser[]" value="ie"> Internet Explore<br>
  <input type="checkbox" name="browser[]" value="moz" checked> Firefox<br>
  <input type="submit">
</form>
```



# The Input Element Radio Button <input>

- Allows the user to select **exactly ONE** from a group of predetermined items
- Each radio button in a group is given the same name and a unique value
- Common Attributes:
  - **type="radio"**
  - name
  - id
  - checked
  - value

## Sample Radio Buttons

Select your favorite browser:

- ☐ Internet Explorer
- ☐ Firefox
- ☐ Opera

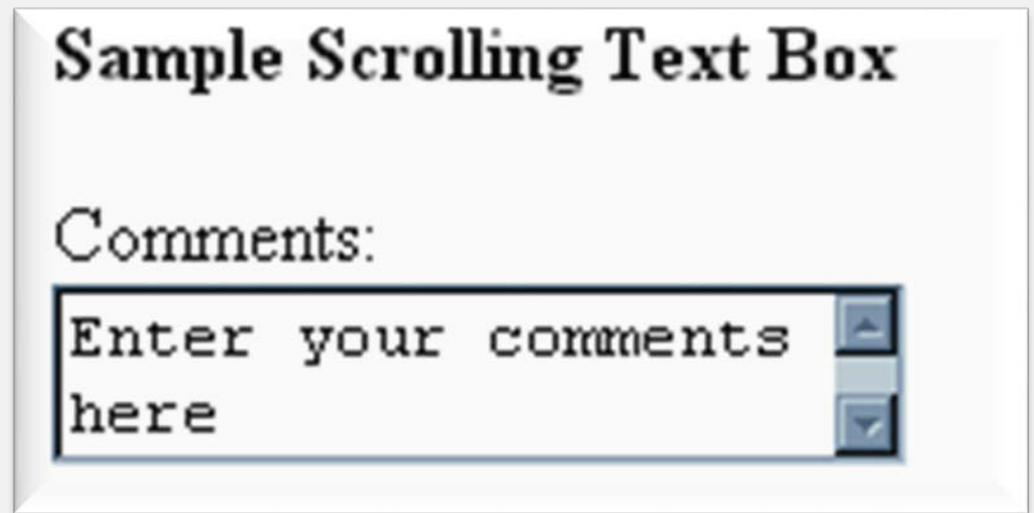
# The Textarea Element <textarea>

- Configures a **scrolling text box**

```
<textarea name="comments" id="comments" rows="4" cols="40">  
</textarea>
```

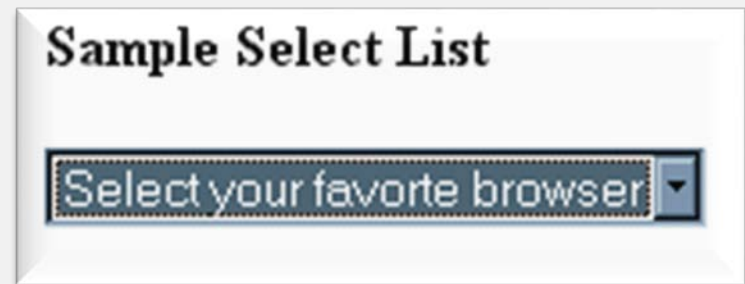
- Common Attributes:

- name
- id
- cols
- rows



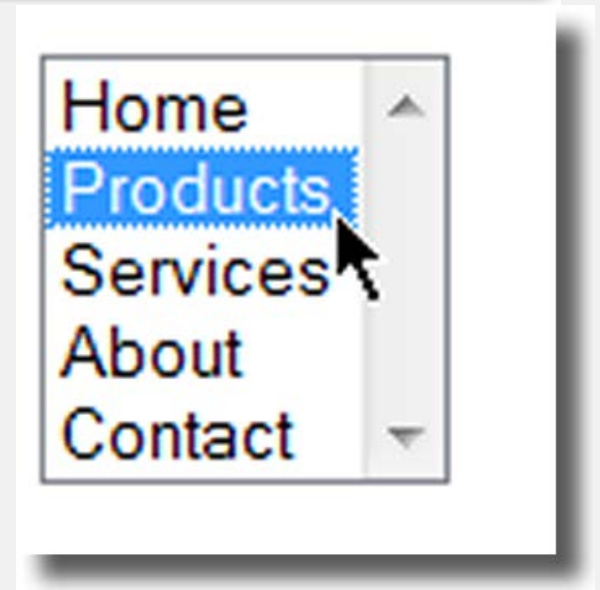
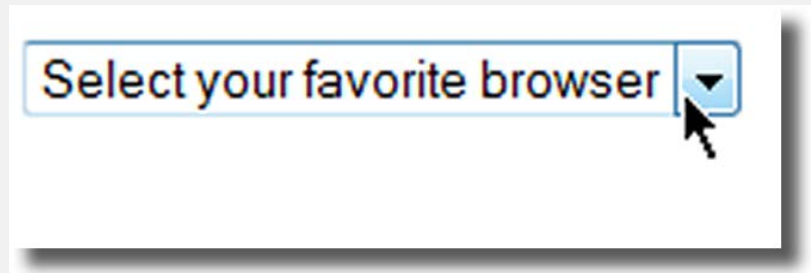
# The Select Element <select>

- Configures a **select list** (along with <option> tags)
- Also known as: Select Box, Drop-Down List, Drop-Down Box, and Option Box.
- Allows the user to select one or more items from a list of predetermined choices.
- Common Attributes:
  - name
  - id
  - size
  - multiple



# The Option Element <option>

- Configures the **options** in a Select List
- Attributes:
  - value
  - selected

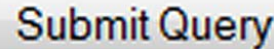


```
<select>  
  <option value="Red">Red</option>  
  <option value="Green">Green</option>  
  <option value="Blue">Blue</option>  
</select>
```

# The Input Element Submit Button <input>

- Submits the form information
- When clicked:
  - Triggers the action method on the <form> tag
  - Sends the form data (the **name=value** pair for each form element) to the web server.
- Attributes:
  - **type="submit"**
  - name
  - id
  - value
- eg. `< input type = "submit" >`

## Sample Submit Button

A rectangular button with a light gray background and a thin black border. The text "Submit Query" is centered on the button in a dark gray, sans-serif font.

# The Input Element Reset Button <input>

➤ Resets the form fields to their initial values

➤ Attributes:

- type="reset"
- name
- id
- value



➤ eg. < input type = "reset" >

# The Input Element Hidden Field <input>

---

- This form control is *not displayed* on the Web page.
- Hidden form fields
  - Can be accessed by both client-side and server-side scripting
- Common Attributes:
  - *type= "hidden"*
  - name
  - id
  - value

# The Label Element <label>

- Associates a **text label** with a form control
- Two Different Formats:

```
<label>Email: <input type="text" name="CustEmail" id="CustEmail"></label>
```

Or

```
<label for="email">Email: </label>  
<input type="text" name="CustEmail" id="email">
```



# The Fieldset & Legend Elements

## ➤ The Fieldset Element

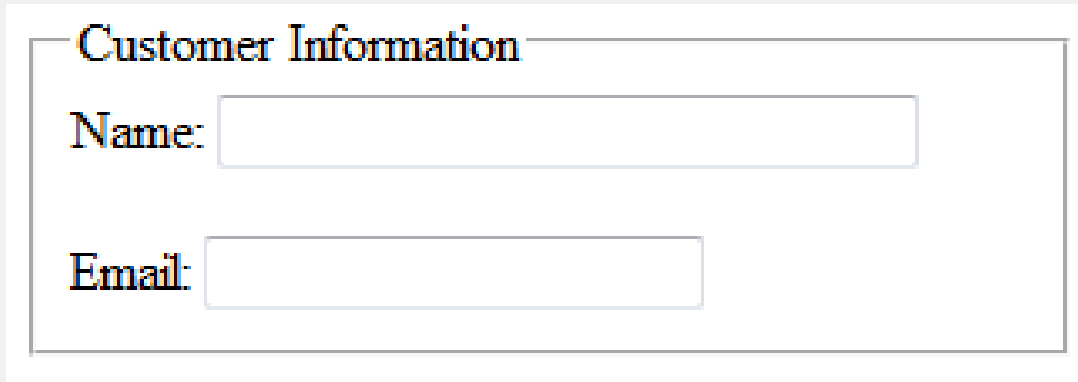
### <fieldset>

- Container tag
- Creates a visual group of form controls on a web page

## ➤ The Legend Element

### <legend>

- Container tag
- Creates a text label within the fieldset



The image shows a web form element. It consists of a rectangular box with a thin border. Inside the box, at the top, is the text "Customer Information" in a bold, black, sans-serif font. Below this text, there are two text input fields. The first field is preceded by the label "Name:" and the second field is preceded by the label "Email:". Both labels and fields are in a black, sans-serif font. The input fields are empty and have a light gray border.

```
<fieldset><legend>Customer Information</legend>
```

```
<label>Name:
```

```
<input type="text" name="CName" id="CName" size="30"></label> <br><br>
```

```
<label>Email:
```

```
<input type="text" name="CEmail" id="CEmail"></label>
```

```
</fieldset>
```

# Using CSS to Style a Form

- Use a **table** to format the form but configure styles instead of HTML table attributes.

```
table { background-color: #eaeaea;  
        width: 20em;  
        font-family: Arial, sans-serif; }  
td { padding: 5px; }  
label { display: block;  
        text-align: right; }
```



The screenshot shows a Windows Internet Explorer browser window titled "Contact Form - Windows Internet Explorer". The address bar shows the local file path "C:\Users\DrMorris\". The Favorites bar includes "Contact Form". The main content area displays a form titled "Contact Us" with the following fields:

- Name:
- E-mail:
- Comments:
- Submit:

# Using CSS to Style a Form

**"Pure" CSS Approach** : Do not use a table to format the form. Use CSS to configure the form elements. Add div selectors when needed.

A diagram showing a form layout within a container labeled 'form'. It contains three rows of labels and input fields. The first two rows have a 'label' and a 'text box'. The third row has a 'label' and a 'scrolling text box'. Below these is a 'submit button'.

A screenshot of a web browser window titled 'Contact Form'. The form is titled 'Contact Us' and contains fields for 'Name:', 'E-mail:', and 'Comments:'. A 'Submit' button is at the bottom.

```
form { background-color:#eaeaea; font-family: Arial, sans-serif;
      width: 350px; padding: 10px; }
label { float: left; clear: left; display: block; width: 100px;
      text-align: right; padding-right: 10px; margin-top: 10px; }
input, textarea { margin-top: 10px; display: block; }
#mySubmit { margin-left: 110px; }
```

# HTML5: Email Text Box <input>

- Accepts text information in **e-mail address format**
- Common Attributes:
  - **type="email"**
  - name
  - id
  - size
  - maxlength
  - value
  - placeholder
  - required



```
<h1>Join Our Newsletter</h1>  
<form method="get">  
  <label for="email">E-mail:</label>  
  <input type="email" name="email" id="email" ><br><br>  
  <input type="submit" value="Sign Me Up!"> <input type="reset">  
</form>
```

# HTML5: Calendar Control <input>

- `<label for="myDate">Choose a Date</label>`  
`<input type="date" name="myDate" id="myDate">`

Form Example - Opera

## Choose A Date

Choose a Date

Send Reset

Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun
48	26	27	28	29	30	1	2
49	3	4	5	6	7	8	9
50	10	11	12	13	14	15	16
51	17	18	19	20	21	22	23
52	24	25	26	27	28	29	30
1	31	1	2	3	4	5	6

Today None

# Practice with an HTML5 Form

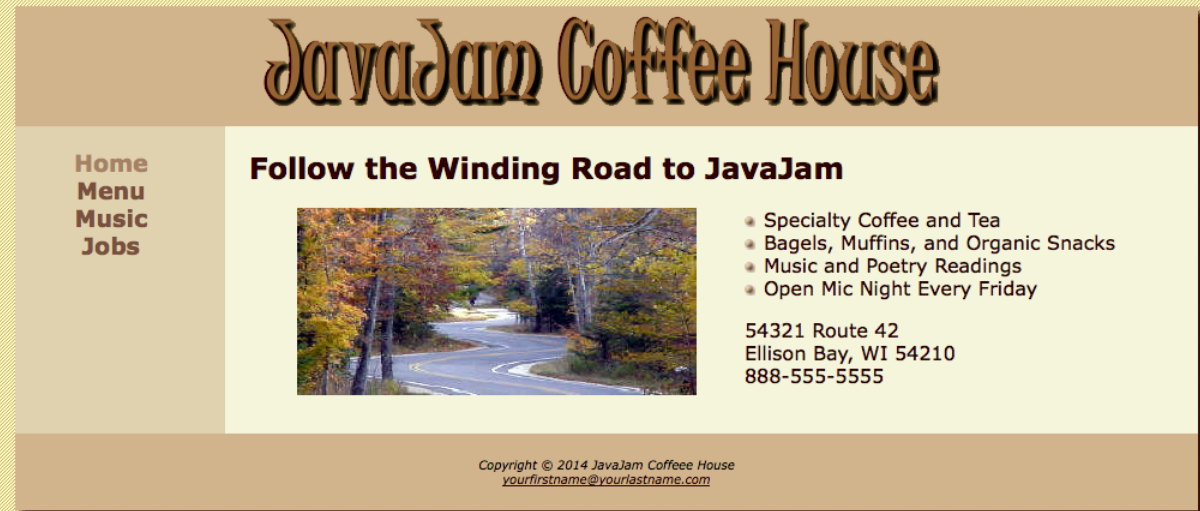
- The form display and functioning varies with browser support.
- Study examples in **Forms.zip**. Note that the action of the form is to invoke either **show\_get.php** or **show\_post.php** which displays in the browser the variables and values passed to the php scripts. This shows those variables passed from the client-side to the server-side.
- The two php scripts shows how you can debug your web applications when you need to know what has been sent from the client computer to the server.

+++++

- Copy the source codes of all the exercises for tables and all examples/exercises to **week04** folder for progress assessments.
- More about CSS3: <http://www.w3schools.com/>

## Case Study (Part 2) – HTML5 + CSS3 + Table & Form

- Copy all the files used in Case Study (Part 1) to a new folder named **CaseStudyPt2**, under the **week04** web folder.
- Add in the necessary codes to create a web page similar to the one below. **The header contains an image.** The **left column has four menu items**. This page is the **“Home Page”**. You can use any image downloaded from a website, but you need to write a comment line in the source code to state where you get the image from.





# Case Study (Part 2) – HTML5 + CSS3 + Table & Form

- When the menu item “[Menu](#)” is clicked, the following page will be shown.
- The actual content in the page is not important, but the [CSS styling is important](#). You can choose to use similar fonts.





# Case Study (Part 2) – HTML5 + CSS3 + Table & Form

- When the menu item “**Music**” is clicked, the following page will be shown.
- The actual content in the page is not important. **The embedded player is for playing MP3 music or MP4 video.** You can use images downloaded from website but must state where you get the images from.



## Case Study (Part 2) – HTML5 + CSS3 + Table & Form

- When the menu item “[Jobs](#)” is clicked, the following page will be shown.
- The [CSS styling of the form is important](#). The action of the form should invoke the [show\\_get.php](#) or [show\\_post.php](#). Note that you must [visit this page through the web server](#).



The screenshot shows a web page for "JavaJam Coffee House". On the left is a vertical menu with links: Home, Menu, Music, and Jobs. The main content area is titled "Jobs at JavaJam" and contains a form for job applications. The form includes fields for Name, E-mail, Start Date, and Experience, with an "Apply Now" button. A copyright notice at the bottom reads: "Copyright © 2014 JavaJam Coffee House yourfirstname@yourlastname.com".

**JavaJam Coffee House**

**Jobs at JavaJam**

Want to work at JavaJam? Fill out the form below to start your application. Required fields are marked with an asterisk \*

\*Name:

\*E-mail:

Start Date:

\*Experience:

Copyright © 2014 JavaJam Coffee House  
yourfirstname@yourlastname.com